



UNITED STATES PATENT AND TRADEMARK OFFICE

70

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/023,378	12/12/2001	Brian Holtz	0007056-0225/P5926	2449
58328	7590	12/16/2005	EXAMINER	
SONNENSCHN NATH & ROSENTHAL LLP FOR SUN MICROSYSTEMS P.O. BOX 061080 WACKER DRIVE STATION, SEARS TOWER CHICAGO, IL 60606-1080			CHOW, CHIH CHING	
			ART UNIT	PAPER NUMBER
			2192	
DATE MAILED: 12/16/2005				

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	10/023,378	HOLTZ ET AL.	
	Examiner	Art Unit	
	Chih-Ching Chow	2192	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --
Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 23 September 2005.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-30 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-30 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 23 September 2005 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. _____.
 3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|--|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152) |
| 3) <input checked="" type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date <u>4/16/04</u> . | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

1. This action is responsive to amendment dated September 23, 2005.
2. Per Applicants' request, the Specification, Drawing (Figure 9), claims 1, 11, and 21 have been amended.
3. Claims 1-30 remain pending.

Response to Amendment

4. Applicants' amendment dated 09/23/2005, responding to the 06/24/2005 Office action provided in the objection of Specification. The examiner has reviewed the updated specification respectfully. The objection to the specification is hereby withdrawn in view of Applicants' amendment to the specification.
5. Applicants' amendment dated 09/23/2005, responding to the 06/24/2005 Office action provided in the objection of drawings. The examiner has reviewed the updated drawing, Figure 9 respectfully. The drawings filed concurrently with the above-mentioned amendment is accepted by the Examiner.

Response to Arguments

6. Applicant's arguments filed 09/23/2005 have been fully considered but they are not persuasive. The applicant's argument on page 15, 1st paragraph, "Beizer fails to teach that the more than one unresolved independent conflicts are 'file tree conflicts determined from a comparison of the first and second file trees,' as claim 1 now recites. In Beizer, conflicts are determined by a comparison of data values. See Figure 5 ('value to '400' from '200') and column 3, lines 6-11 of Beizer. Thus, the conflicts in Beizer are data element conflicts, not file tree conflicts, and they are determined by a data element per data element comparison, not by a comparison of file trees.
7. Examiner's Response: Beizer also compares two file trees, see Beizer's FIG. 3b, two file trees containing unresolved independent conflicts are presented, user1 changes

and user2 changes (*first and second file trees*). Comparing the data values is just a way of comparing two files, it's well known to the people in the art how to compare two files or even two file directories (file trees) without comparing data values inside of the file, e.g. using 'chksum' (checksum) is one of the common way to do it. See 35 USC § 102 rejections (claims include the amendments) herein below:

Claim Rejections - 35 USC § 102

8. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(a) the invention was known or used by others in this country, or patented or described in a printed publication in this or a foreign country, before the invention thereof by the applicant for a patent.

9. Claims 1-30 are rejected under 35 U.S.C. 102(a) as being anticipated by US Patent No. 6,240,414 by Mordechai M. Beizer et al. (hereinafter "Beizer").

CLAIM

1. A method for processing conflicts in a first and second file tree comprising:
presenting one or more conflicts to a user;
presenting more than one unresolved independent conflicts as a single conflict to said user, wherein said more than one unresolved independent conflicts are file tree conflicts determined from a comparison of the first and second file trees;
obtaining input from said user; and
handling said one or more conflicts and said single conflict based on said input.

Beizer

Beizer teaches resolving conflicts in a shared data environment where a plurality of users can concurrently access files. See Beizer column 9, lines 35-52, "according to the invention, the affected user(s) are **notified that an embedded content conflict has occurred** (*presenting one or more conflicts to a user*) but that resolution of the conflict deferred. In the most preferred embodiment, the conflicting documents are both stored in the WorkFolder as alternate 'sub-items', where each sub-item is one version of the actual document content or filled placeholder. When a WorkFolder's contents is **displayed** (*presenting*), the sub-items can be grouped together and **identified as**

representing conflicting alternatives (*presenting more than one unresolved independent conflicts to user*). This may be done automatically by providing sub-items with a special 'in conflict' attribute tag. Preferably, any user with the appropriate modify rights can **resolve this conflict by explicitly selecting which of the conflicting content documents should be preserved** (*obtaining input from user*). Preferably, the unselected sub-item is then automatically deleted. The conflict also automatically resolves itself if all but one of the conflicting sub-items are deleted or moved elsewhere in the contents (*handling said one or more conflicts and said single conflict based on said input*). For the 'comparison of the first and second file trees', see Beizer's FIG. 3b, two file trees containing unresolved independent conflicts are presented, user1 changes and user2 changes (*first and second file trees*); and column 5, line 65 to column 6, line 9, "When the local edited WorkFolder 58 is saved, it can be compared to the local **original copy 54** to (*comparing the first and the second file trees*) determine what changes were made by the user and also compared to the master copy 52, which may itself have been changed by a **second user in the interim.**" And "According to the present invention, when a user edits the local copy 58 and then saves the edited WorkFolder (thus updating their local "original" copy 54), the reconciliation program **compares** the version of the most recently saved local copy 54 to the master version of the WorkFolder 52 on the server." Further the disclosure in column 7, line 23 to column 8 line 31; and column 9, lines 54-59, "In some cases, such as when

the conflicting documents are known to have the same format, a specialized **document compare routine** (either integral with the reconciliation system or part of an external software package, such as a word processor) is used to **perform a comparison of the documents and list the document conflicts for resolution by the user.**” – It’s well known to the people in the art how to compare two files or even two file directories, e.g. using checksum is one of a common way.

2. The method of claim 1 wherein one or more winning conflicts are applied to a file tree with which they were in conflict.

For the feature of claim 1 see claim 1 rejection. The ‘file tree’ is actually a file structure (it’s well-known for the people in the art to structure files in a ‘file tree’ structure, such as to group files into directory of files), which is also covered in Beizer’s disclosure, see Beizer’s column 1, lines 30-34, “Documents and data of this type may be stored in a **shared, structured data object**. These objects generally comprise various sections into which related documents are grouped (*structured files grouping as a file tree*), one or more forms into which data can be entered, links to other documents, etc. The data object is stored in a manner which allows it to be simultaneously accessed by many different users as needed.”

3. The method of claim 2 wherein said winning conflicts are server operations.

For the feature of claim 2 see claim 2 rejection. In Beizer’s disclosure, the winning conflicts can be either server or client operations, it depends whichever is selected by the user. See Beizer’s FIG. 3b and 3c.

4. The method of claim 2 wherein said winning conflicts are client operations.

Same as claim 3 rejection.

5. The method of claim 3 wherein said server operations are translated back up a conflict list across all previous server operations, and then down a conflict list across all client operations.

For the feature of claim 3 see claim 3 rejection. Beizer's teaching also distributes the winning conflict to all the clients, see Beizer's column 4, lines 27-30, "The WorkFolder data object is stored persistently in such a manner that it can be **stored, forwarded, and distributed across a network** and be accessed and modified by software running on a variety of different platforms." Also see the description of Figure 4, Beizer's column 13, lines 19-64, **a series of steps are performed for the replication** of the reconciled files, the updates are done at all the effected sites.

6. The method of claim 4 wherein said client operations are translated back up a conflict list across all previous client operations, and then down the conflict list across all server operations.

See claim 3, 4 and 5 rejections.

7. The method of claim 5 wherein if said translation is a rename or a reparent of an object of said winning operation, said winning operation is translated to refer to said object using its new lineage.

For the feature of claim 5 see claim 5 rejection. In Beizer's reconciliation process, the winning operation is rename/reparent, and a new lineage is thus applied to the winning file.

8. The method of claim 6 wherein if said translation is a rename or a reparent of one of an ancestor of said object in the tree, said winning operation is translated to refer to the object using its new lineage.

For the feature of claim 6 see claim 6 rejection. For the rest of claim 8 feature, see claim 7 rejection.

9. The method of claim 3 wherein said winning server operations are en-queued for transmission to the client.

For the feature of claim 3 see claim 3 rejection. See Claim 5, a series steps (*en-queued*) are performed in order to replicate reconciled file.

10. The method of claim 4 wherein said

For the feature of claim 4 see claim 4

winning client operations are applied against said server's filesystem.

11. An article of manufacture comprising:
a computer usable medium having computer readable program code embodied therein for processing conflicts in a first and second file tree, said computer readable program code in said article of manufacture comprising:

computer readable program code configured to cause said computer to present one or more conflicts to a user;

computer readable program code configured to cause said computer to present more than one unresolved independent conflicts as a single conflict to said user, wherein said more than one unresolved independent conflicts are file tree conflicts determined from a comparison of the first and second file trees;

computer readable program code configured to cause said computer to obtain input from said user; and

computer readable program code configured to cause said computer to handle said one or more conflicts and said single conflict based on said input.

12. The article of manufacture of claim 11 wherein one or more winning conflicts are applied to a file tree with which they are in conflict.

13. The article of manufacture of claim 12 wherein said winning conflicts are server operations.

14. The article of manufacture of claim 12 wherein said winning conflicts are client

rejection. For the rest of claim 10 feature, see claim 1, 3, and 5 rejections.

Beizer's teaching is an article of manufacture comprising a computer useable medium, computer readable program code, see Beizer's FIG. 2. For the rest of claim 11 feature see claim 1 rejection. For the 'comparison of the first and second file trees', see claim 1 rejection.

For the feature of claim 11 see claim 11 rejection. For the rest of claim 12 feature, see claim 2 rejection.

For the feature of claim 12 see claim 12 rejection. For the rest of claim 13 feature, see claim 3 rejection.

For the feature of claim 12 see claim 12 rejection. For the rest of claim 14 feature,

operations.

see claim 4 rejection.

15. The article of manufacture of claim 13 wherein said server operations are translated back up a conflict list across all previous server operations, and then down a conflict list across all client operations.

For the feature of claim 13 see claim 13 rejection. For the rest of claim 15 feature, see claim 5 rejection.

16. The article of manufacture of claim 14 wherein said client operations are translated back up a conflict list across all previous client operations, and then down the conflict list across all server operations.

For the feature of claim 14 see claim 14 rejection. For the rest of claim 16 feature, see claim 6 rejection.

17. The article of manufacture of claim 15 wherein if said translation is a rename or a reparent of an object of said winning operation, said winning operation is translated to refer to said object using its new lineage.

For the feature of claim 15 see claim 15 rejection. For the rest of claim 17 feature, see claim 7 rejection.

18. The article of manufacture of claim 16 wherein if said translation is a rename or a reparent of one of an ancestor of said object in the tree, said winning operation is translated to refer to the object using its new lineage.

For the feature of claim 16 see claim 16 rejection. For the rest of claim 18 feature, see claim 8 rejection.

19. The article of manufacture of claim 13 wherein said winning server operations are en-queued for transmission to the client.

For the feature of claim 13 see claim 13 rejection. For the rest of claim 19 feature, see claim 9 rejection.

20. The article of manufacture of claim 14 wherein said winning client operations are applied against said server's filesystem.

For the feature of claim 14 see claim 14 rejection. For the rest of claim 20 feature, see claim 10 rejection.

21. A computer program product comprising:
a computer useable medium having computer readable program code embodied therein configured to process conflicts in a

Beizer's teaching is computer program product comprising: computer useable medium, computer readable program code, see Beizer's FIG. 2. For the rest of claim 21 feature see claim 1 rejection. For the

first and second file tree, said computer program product comprising:

computer readable code configured therein to cause a computer to present one or more conflicts to a user;

computer readable code configured therein to cause a computer to present more than one unresolved independent conflicts as a single conflict to said user, wherein said more than one unresolved independent conflicts are tile tree conflicts determined from a comparison of the first and second file trees;

computer readable code configured therein to cause a computer to obtain input from said user; and

computer readable code configured therein to cause a computer to handle said one or more conflicts and said single conflict based on said input.

'comparison of the first and second file trees', see claim 1 rejection.

22. The computer program product of claim 21 wherein one or more winning conflicts are applied to a file tree with which they were in conflict.

For the feature of claim 21 see claim 21 rejection. For the rest of claim 22 feature, see claim 2 rejection.

23. The computer program product of claim 22 wherein said winning conflicts are server operations.

For the feature of claim 22 see claim 22 rejection. For the rest of claim 23 feature, see claim 3 rejection.

24. The computer program product of claim 22 wherein said winning conflicts are client operations.

For the feature of claim 22 see claim 22 rejection. For the rest of claim 24 feature, see claim 4 rejection.

25. The computer program product of claim 23 wherein said server operations are translated back up a conflict list across all previous server operations, and then down a conflict list across all client operations.

For the feature of claim 23 see claim 23 rejection. For the rest of claim 25 feature, see claim 5 rejection.

- | | |
|---|---|
| 26. The computer program product of claim 24 wherein said client operations are translated back up a conflict list across all previous client operations, and then down the conflict list across all server operations. | For the feature of claim 24 see claim 24 rejection. For the rest of claim 26 feature, see claim 6 rejection. |
| 27. The computer program product of claim 25 wherein if said translation is a rename or a reparent of an object of said winning operation, said winning operation is translated to refer to said object using its new lineage. | For the feature of claim 25 see claim 25 rejection. For the rest of claim 27 feature, see claim 7 rejection. |
| 28. The computer program product of claim 26 wherein if said translation is a rename or a reparent of one of an ancestor of said object in the tree, said winning operation is translated to refer to the object using its new lineage. | For the feature of claim 26 see claim 26 rejection. For the rest of claim 28 feature, see claim 8 rejection. |
| 29. The computer program product of claim 23 wherein said winning server operations are en-queued for transmission to the client. | For the feature of claim 23 see claim 23 rejection. For the rest of claim 29 feature, see claim 9 rejection. |
| 30. The computer program product of claim 24 wherein said winning client operations are applied against said server's filesystem. | For the feature of claim 24 see claim 24 rejection. For the rest of claim 30 feature, see claim 10 rejection. |

Conclusion

10. The following summarizes the status of the claims:

35 USC § 102 rejection: Claims 1-30

11. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

Neeman, US 5,588,147, discloses a method for a replication facility provides for the replication of files or portions of files in a distributed environment. The replication

facility is able to replicate any subtree within a distributed namespace of the distributed environment.

Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire **THREE MONTHS** from the mailing date of this action. In the event a first reply is filed within **TWO MONTHS** of the mailing date of this final action and the advisory action is not mailed until after the end of the **THREE-MONTH** shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than **SIX MONTHS** from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Chih-Ching Chow whose telephone number is 571-272-3693. The examiner can normally be reached on 7:30am - 4:00pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q. Dam can be reached on 571-272-3695. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300. Any inquiry of a general nature of relating to the status of this application should be directed to the **TC2100 Group receptionist: 571-272-2100**.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you

Art Unit: 2192

have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).


Chih-Ching Chow

Examiner

Art Unit 2192

December 09, 2005

CC



TUAN DAM
SUPERVISORY PATENT EXAMINER